Hash Chemistry on a Cellular Grid: An Open-Ended Artificial Chemistry System with Computational Efficiency and Nontrivial Spatio-Temporal Dynamics

Hiroki Sayama[1,2]

[1]Binghamton University, USA
[2]Waseda University, Japan
sayama@binghamton.edu

Open-endedness is currently one of the most actively investigated topics in Artificial Life (Stanley, 2019; Packard et al., 2019; Stepney, 2021; Borg et al., 2023; Stepney and Hickinbotham, 2023) and particularly in the distributed dynamical systems-based ALife research community (Adams et al., 2017; Chan, 2023; Sayama and Nehaniv, 2024). To facilitate open-ended evolution in such ALife models, we previously proposed the concept of "cardinality leap", i.e., a general mathematical operation to drastically increase the cardinality of the possibility space by allowing the formation of higher-order entities. Its effectiveness was demonstrated in "Hash Chemistry" (Sayama, 2019), a hash function-based spatial artificial chemistry model. The original Hash Chemistry was computationally very expensive, and it was also bounded in terms of the complexity growth of self-replicating entities. Recently, these limitations have been addressed in a nonspatial variant of Hash Chemistry based on multisets (Sayama, 2024) that achieved substantial speed-up of model simulations and significantly better demonstration of spontaneous complexity increase and open-endedness. However, this nonspatial version was created at the cost of nontrivial spatial ("ecological") interactions of self-replicating higher-order entities. It remains an open question how to further improve Hash Chemistry so that it can demonstrate nontrivial spatial interactions of replicators without sacrificing high computational efficiency.

In this short abstract, we propose yet another variant of Hash Chemistry that operates in a regular 2D spatial grid (called "Cellular Hash Chemistry" hereafter). We assume a typical cellular automata-like $L \times L$ grid in which each cell can be empty or contain an individual elementary entity of type i ($i \in \{1, \ldots, k\}$). In this study, we used $L = 100$, $k = 1000$, and initial configurations that were almost empty but with 10 randomly generated initial elementary entities scatted over space (Fig. 1, top left). The updating rule of Cellular Hash Chemistry is designed following the random selection and competition procedure adopted in (Sayama, 2024) and proceeds as follows:

1. Randomly choose *four* non-overlapping square regions in the space, whose sizes l_1 and l_2 are randomly chosen from $\{1, \ldots, L/2\}$. Two squares are of size $l_1 \times l_1$ and the other two are of size $l_2 \times l_2$. We used these two different sizes of patterns in order to represent competition among spatial patterns of different sizes.

2. Extract the spatial patterns at those four square regions from the system configuration, and then remove any blank rows and columns from those patterns. This operation, which we call *shrinking*, is implemented so that patterns can be identified more robustly even if there are minor vertical/horizontal translations in the individual entity positions or gaps in the spatial patterns.

3. Calculate the fitnesses of the four shrunk patterns using a hash function. Specifically, the hash value is calculated as $(h(\mathrm{array}) \bmod M)/M$, where $h(x)$ is the hash function[1], array is the spatial pattern, and M is a normalization parameter ($M = 100000000$ in this study).

4. Determine the winning pattern out of the four, and then let the winner copy itself into the other region of the same size (i.e., if the winning pattern is of size $l_1 \times l_1$, copy that pattern into the other region of the same size while leaving the other two regions of size $l_2 \times l_2$ untouched). In copying, introduce random mutations with small probability[2].

The above steps constitute one update of the system configuration. We implemented a simulator in Wolfram Mathematica 14.0 and conducted numerical simulations. Thanks to the extremely simple updating algorithm, simulations of Cellular Hash Chemistry remained highly computationally efficient even though the model captures spatial interactions (e.g., a simulation for 2×10^5 updates with $L = 100$ would complete in less than 10 minutes in an ordinary laptop without any code optimization).

[1]We used Mathematica 14.0's `Hash` function in its default settings.

[2]In this study, each cell within the copied pattern was reset to an empty state with 0.2% probability or mutated to a random value selected from $\{1, \ldots, k\}$ with 0.2% probability.

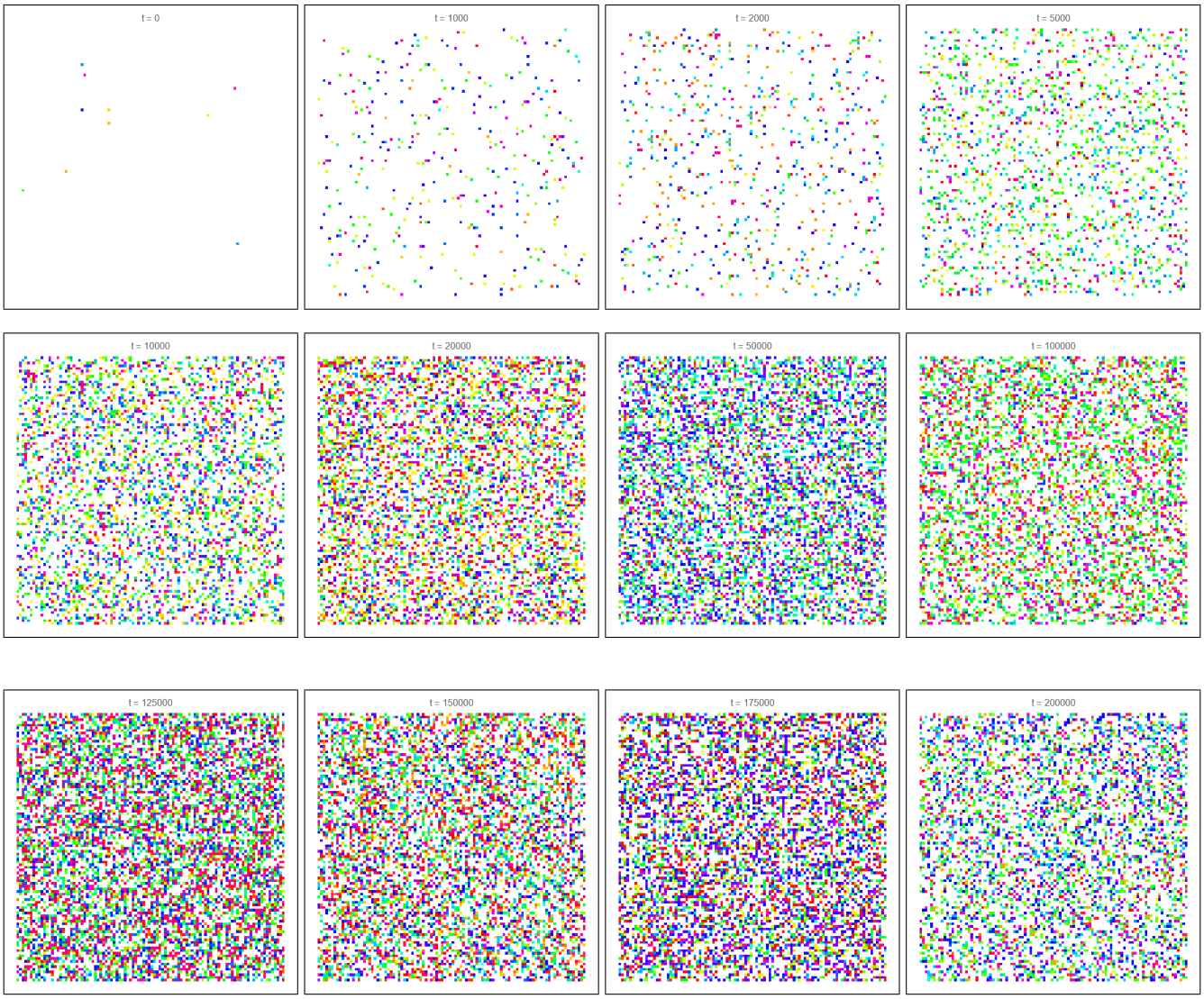

Figure 1: A sample simulation run of Cellular Hash Chemistry. Snapshots of system configurations are arranged temporally from left to right and then top to bottom. The top two rows show the initial population growth process in pseudo-logarithmic time intervals ($t = 0, 1000, 2000, 5000, 10000, 20000, 50000, 100000$), while the bottom row shows the long-term dynamics of the system in linear time intervals ($t = 125000, 150000, 175000, 200000$). Colors represent different element types, and blank (white) spaces represent empty cells. It is observed in these visualizations that the system does not converge to dynamic equilibria, but rather, it keeps dynamically changing indefinitely, not only in terms of system compositions but also the population density and some noticeable microscopic spatial structures (e.g., vertical/horizontal/diagonal microstructures and specific small pattern blocks that are present at multiple locations in the space).

Numerical simulations demonstrated that Cellular Hash Chemistry successfully achieved long-term evolution with nontrivial spatio-temporal dynamics with no significant computational burden. Figure 1 shows an illustrative sample simulation run, in which one can observe that the system kept dynamically changing indefinitely without falling into dynamic equilibria, continuously changing system compositions and the population density, and even generating some noticeable microscopic spatial structures from time to time.

We hope Cellular Hash Chemistry is an insightful and practically useful new addition to the Hash Chemistry model family. It combines desirable aspects of both of the previous two models – nontrivial spatial interactions from the original model (Sayama, 2019) and computational efficiency from the recent nonspatial variant (Sayama, 2024). A systematic computational investigation of the evolutionary properties of this new system is currently underway and will be reported at the conference and in a publication elsewhere.

References

Adams, A., Zenil, H., Davies, P. C., and Walker, S. I. (2017). Formal definitions of unbounded evolution and innovation reveal universal mechanisms for open-ended evolution in dynamical systems. *Scientific Reports*, 7(1):997.

Borg, J. M., Buskell, A., Kapitany, R., Powers, S. T., Reindl, E., and Tennie, C. (2023). Evolved open-endedness in cultural evolution: A new dimension in open-ended evolution research. *Artificial Life*, pages 1–22.

Chan, B. W.-C. (2023). Towards large-scale simulations of open-ended evolution in continuous cellular automata. In *Proceedings of the Companion Conference on Genetic and Evolutionary Computation*, pages 127–130.

Packard, N., Bedau, M. A., Channon, A., Ikegami, T., Rasmussen, S., Stanley, K. O., and Taylor, T. (2019). An overview of open-ended evolution: Editorial introduction to the open-ended evolution II special issue. *Artificial Life*, 25(2):93–103.

Sayama, H. (2019). Cardinality leap for open-ended evolution: Theoretical consideration and demonstration by Hash Chemistry. *Artificial Life*, 25(2):104–116.

Sayama, H. (2024). Non-spatial Hash Chemistry as a minimalistic open-ended evolutionary system. In *Proceedings of the 2024 IEEE Congress on Evolutionary Computation (CEC 2024)*. IEEE. In press; preprint available at `https://arxiv.org/abs/2404.18027`.

Sayama, H. and Nehaniv, C. L. (2024). Self-reproduction and evolution in cellular automata: 25 years after evoloops. *arXiv preprint arXiv:2402.03961*.

Stanley, K. O. (2019). Why open-endedness matters. *Artificial Life*, 25(3):232–235.

Stepney, S. (2021). Modelling and measuring open-endedness. *Artificial Life*, 25(1):9.

Stepney, S. and Hickinbotham, S. (2023). On the open-endedness of detecting open-endedness. *Artificial Life*, pages 1–26.